

FIG. 1

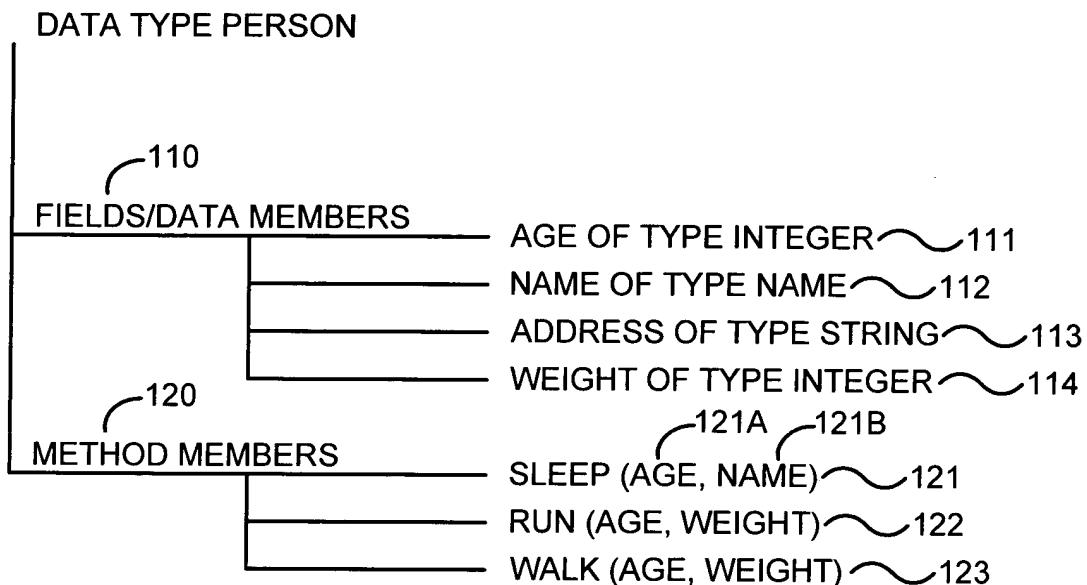


FIG. 2

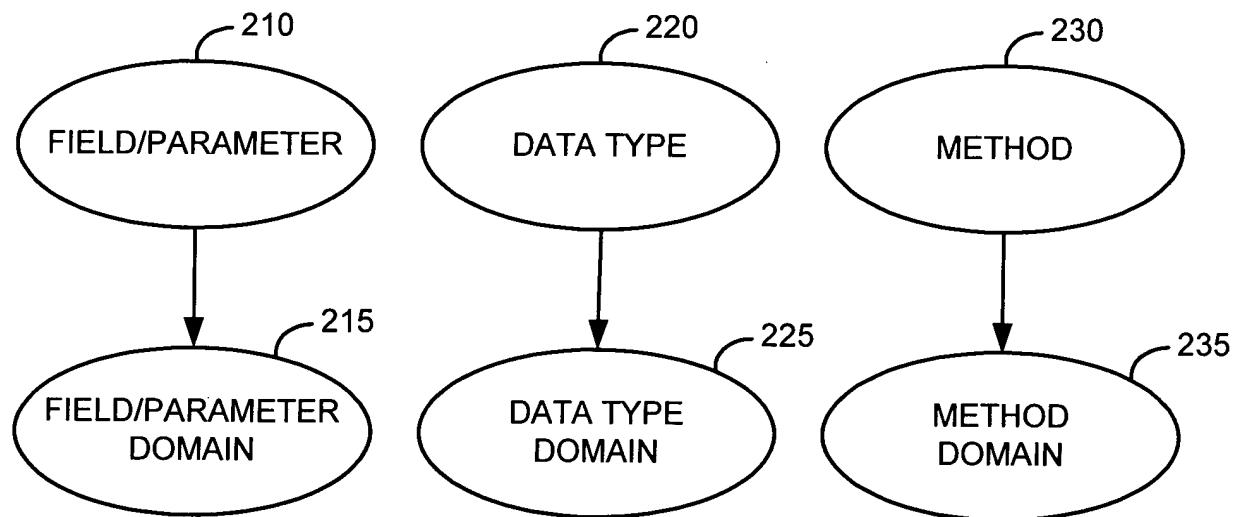


FIG. 3A

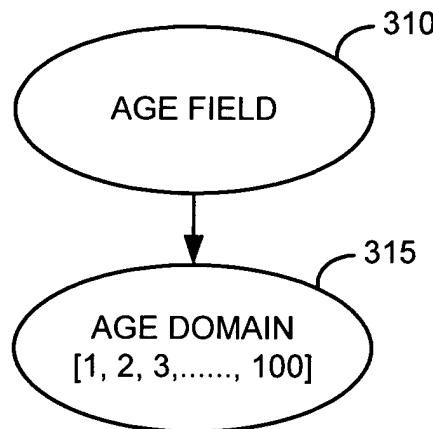


FIG. 3B

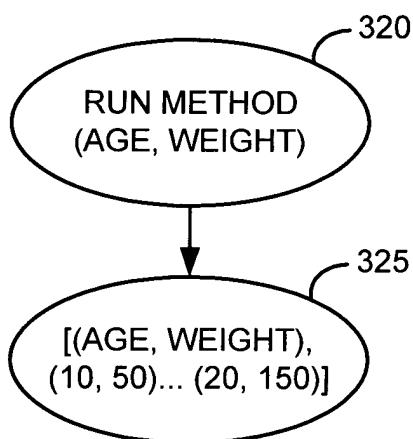


FIG. 3C

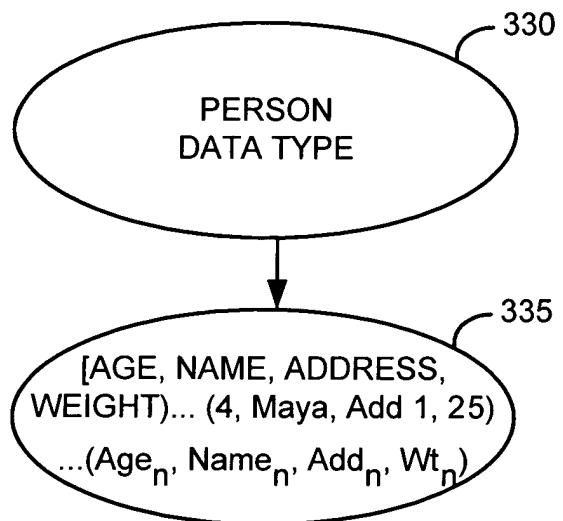


FIG. 4

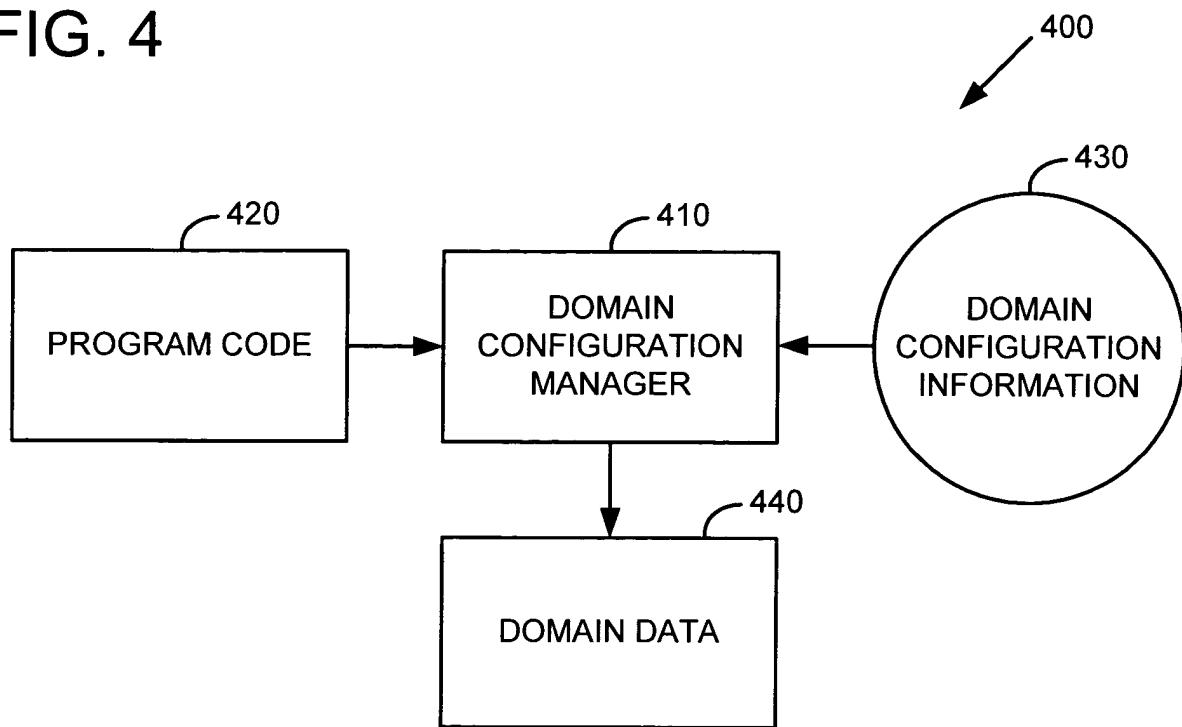


FIG. 6

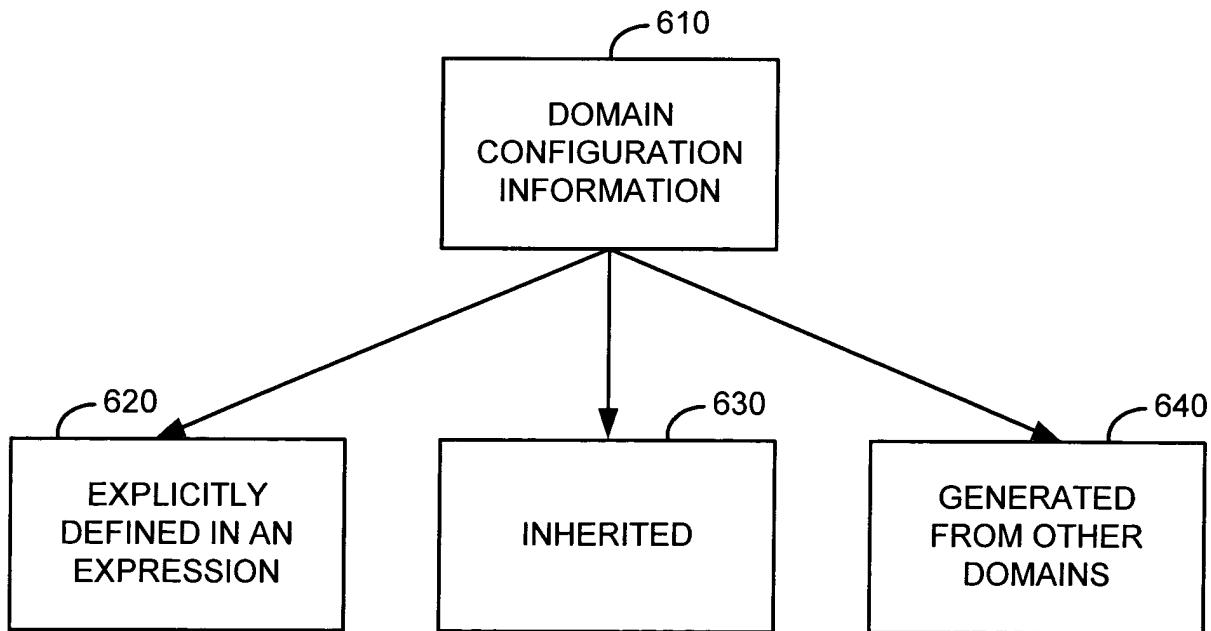


FIG. 5

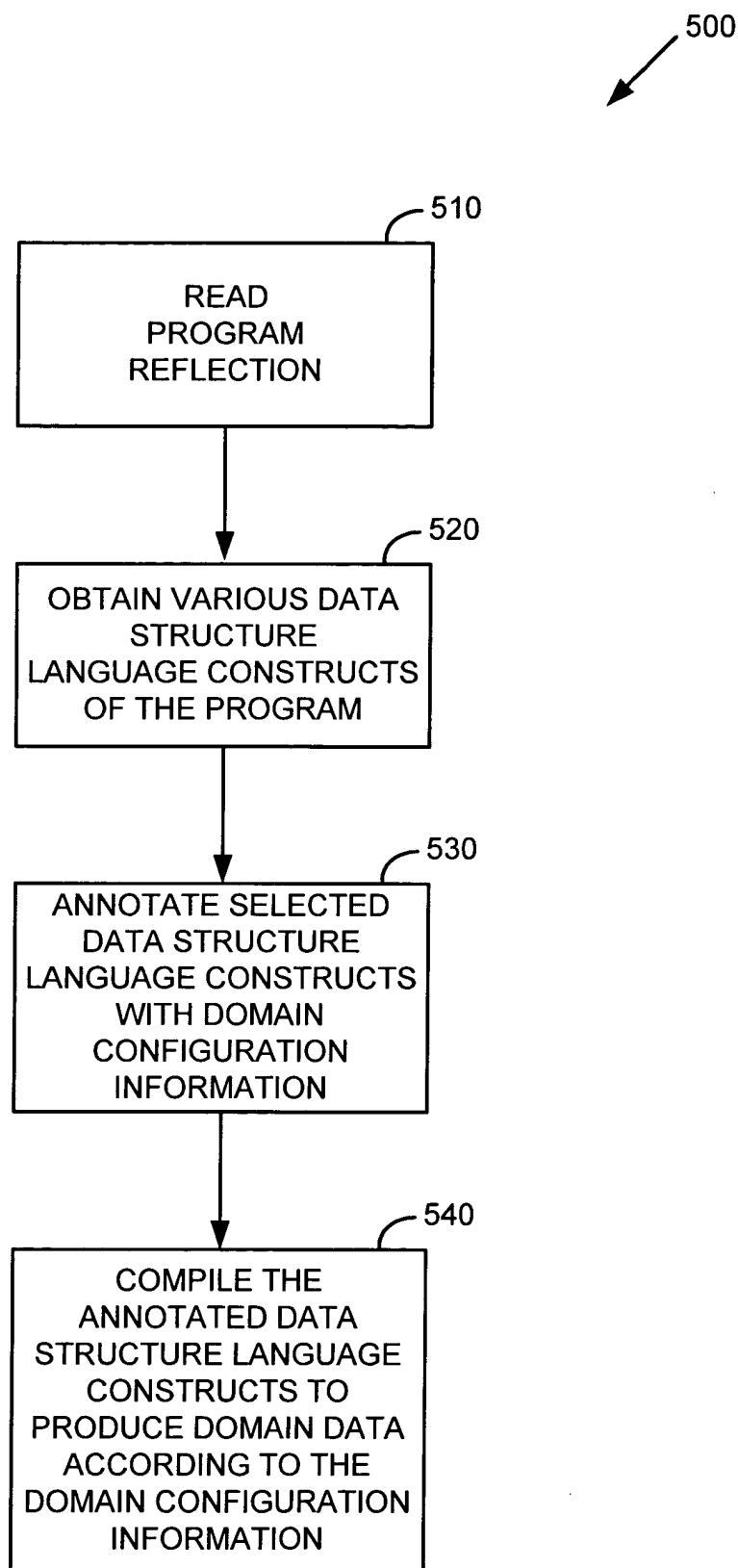


FIG. 7A

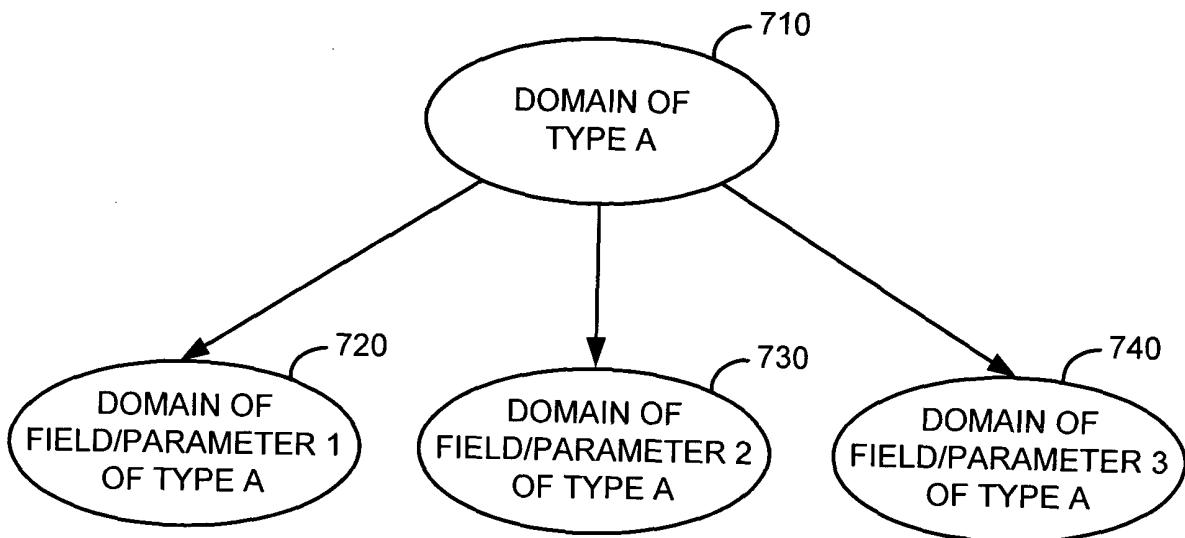


FIG. 7B

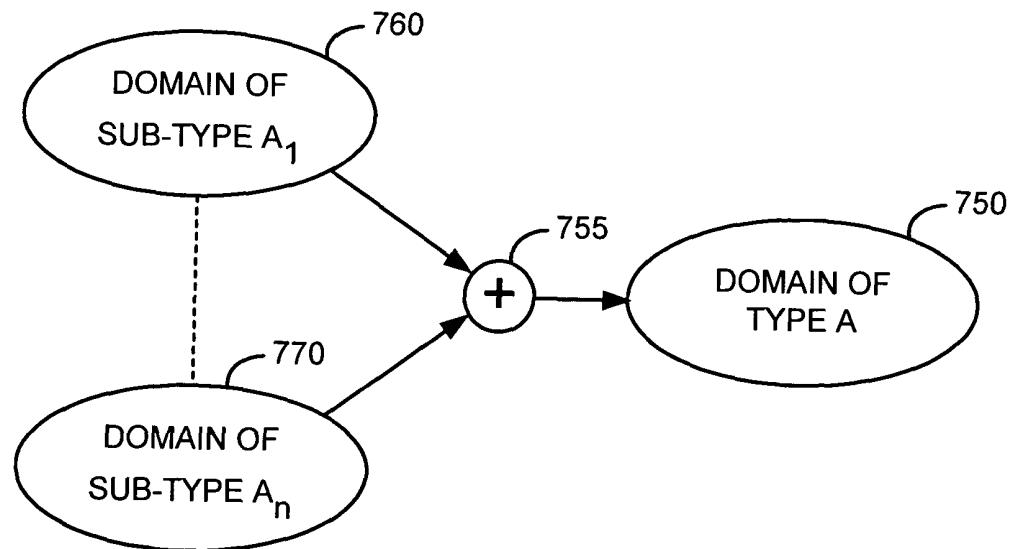


FIG. 8A

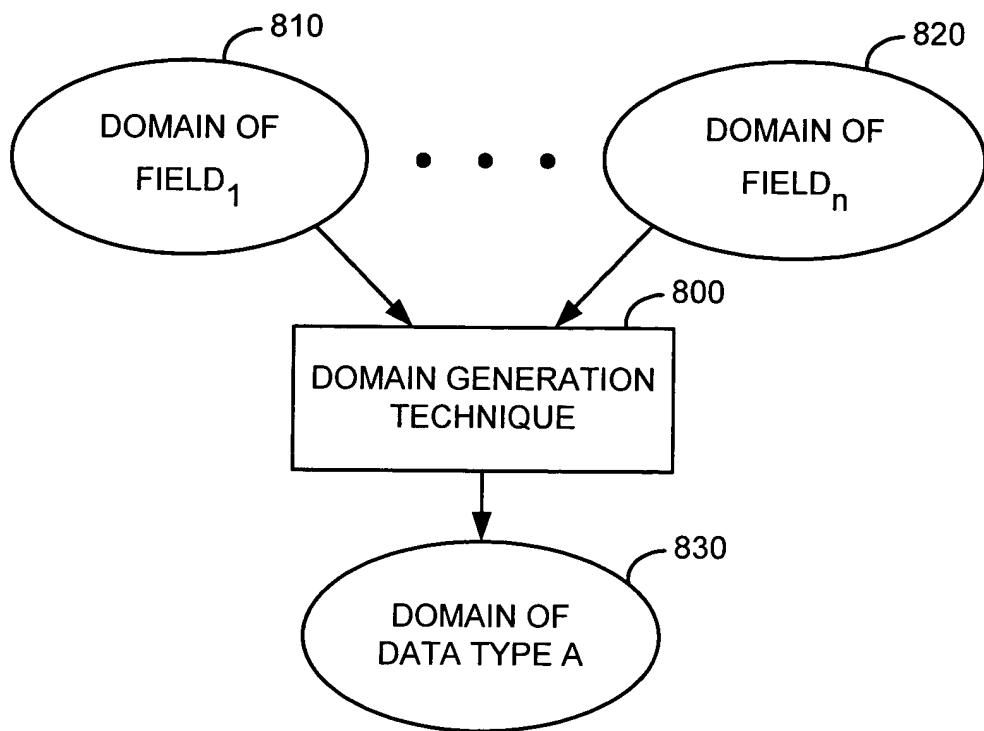


FIG. 8B

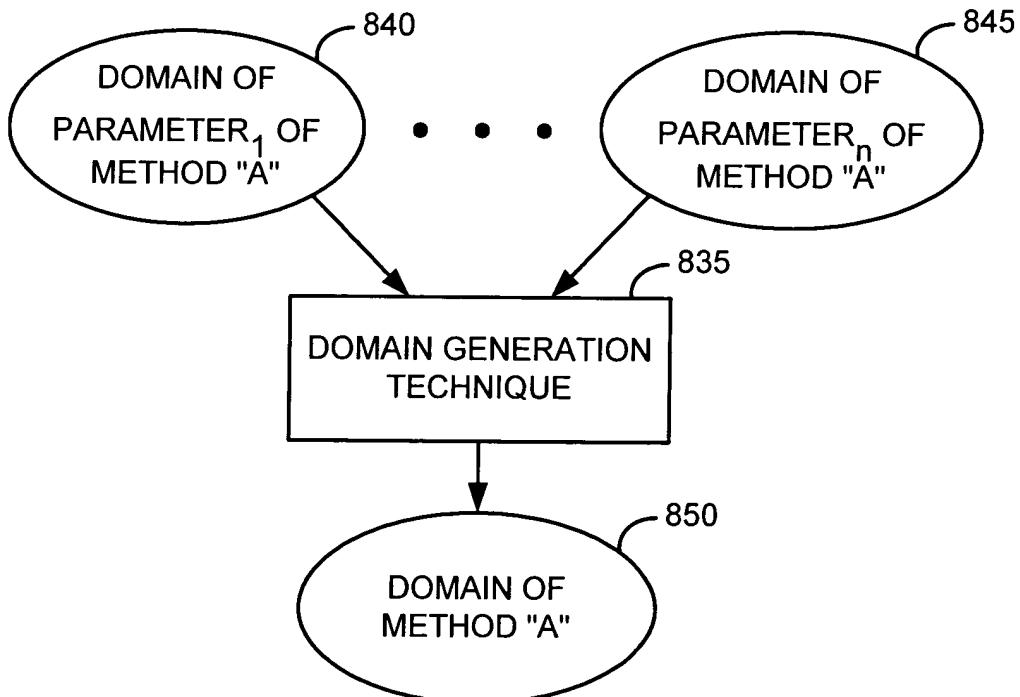


FIG. 9

```
[ProfilerHint(FieldReadProfileable=true)]
abstract structure Exp
  case Const ~~~~~910
  val as Integer ~~~~~911
  case Bin ~~~~~920
  op as Op ~~~~~921
  left as Exp ~~~~~922
  right as Exp ~~~~~923
  case Let ~~~~~930
  name as Name ~~~~~931
  def as Exp ~~~~~932
  body as Exp ~~~~~933
  case Var ~~~~~940
  name as Name ~~~~~941

enum Op
  Add ~~~~~950
  Sub

type Name = String ~~~~~960
```

900
→

FIG. 10

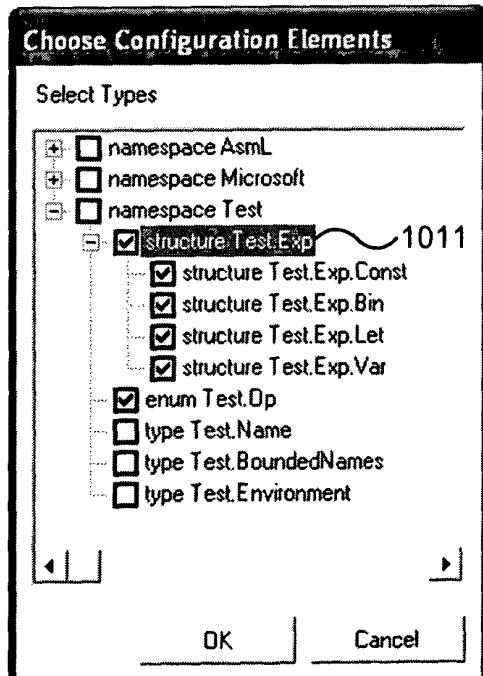


FIG. 11

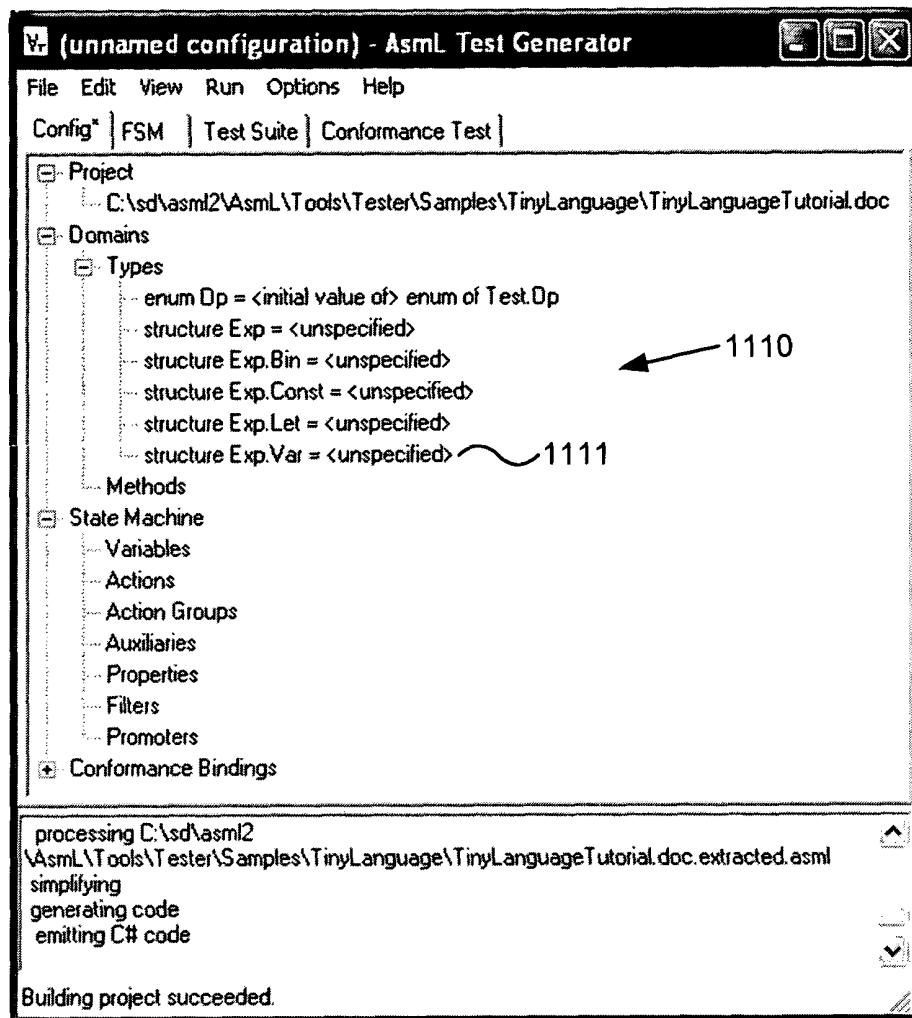


FIG. 12

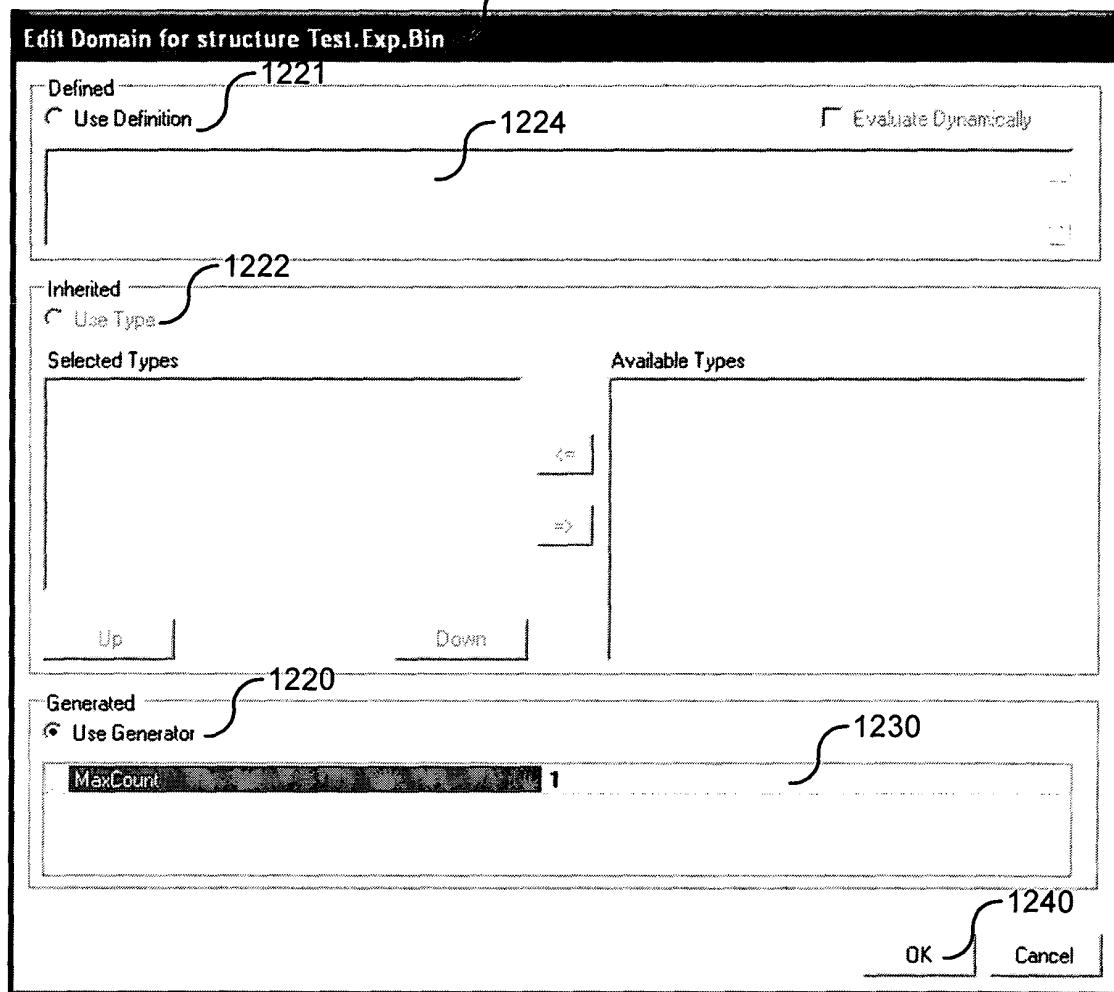


FIG. 13

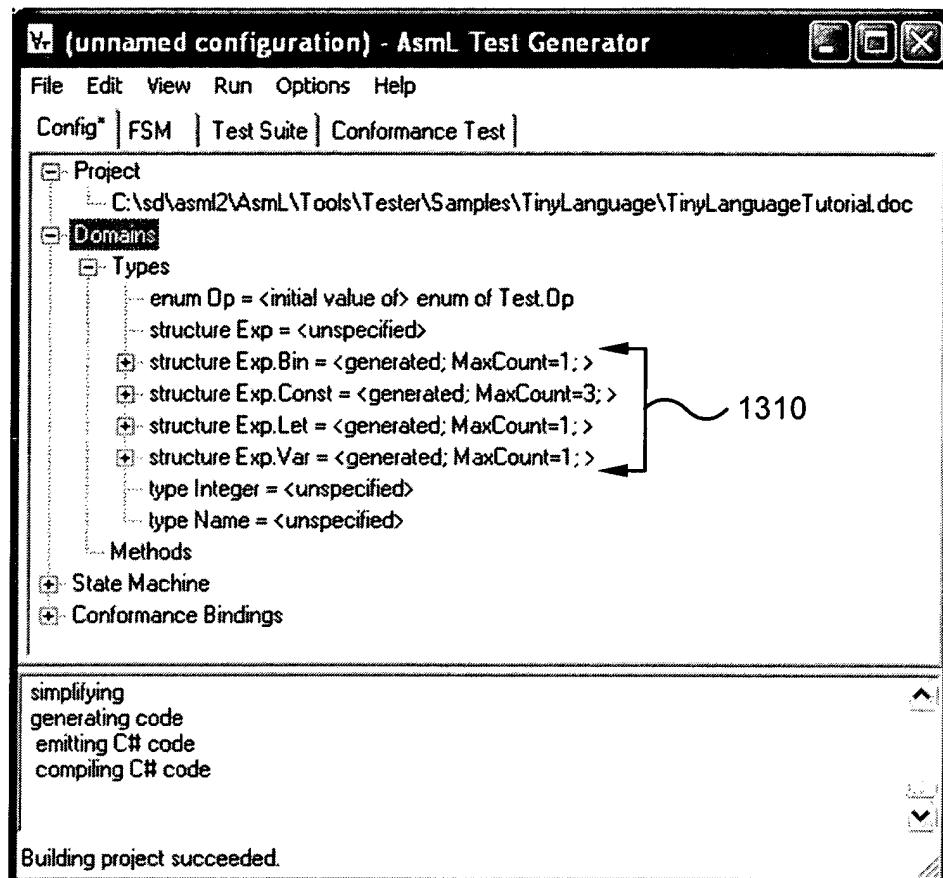


FIG. 14

Edit Domain for structure Test.Exp

Defined

Use Definition Evaluate Dynamically

Inherited

Use Type 1410 1420

Selected Types

structure Test.Exp.Const
structure Test.Exp.Bin
structure Test.Exp.Var
structure Test.Exp.Let

Available Types

Up Down

Generated

Use Generator

MaxCount

OK Cancel

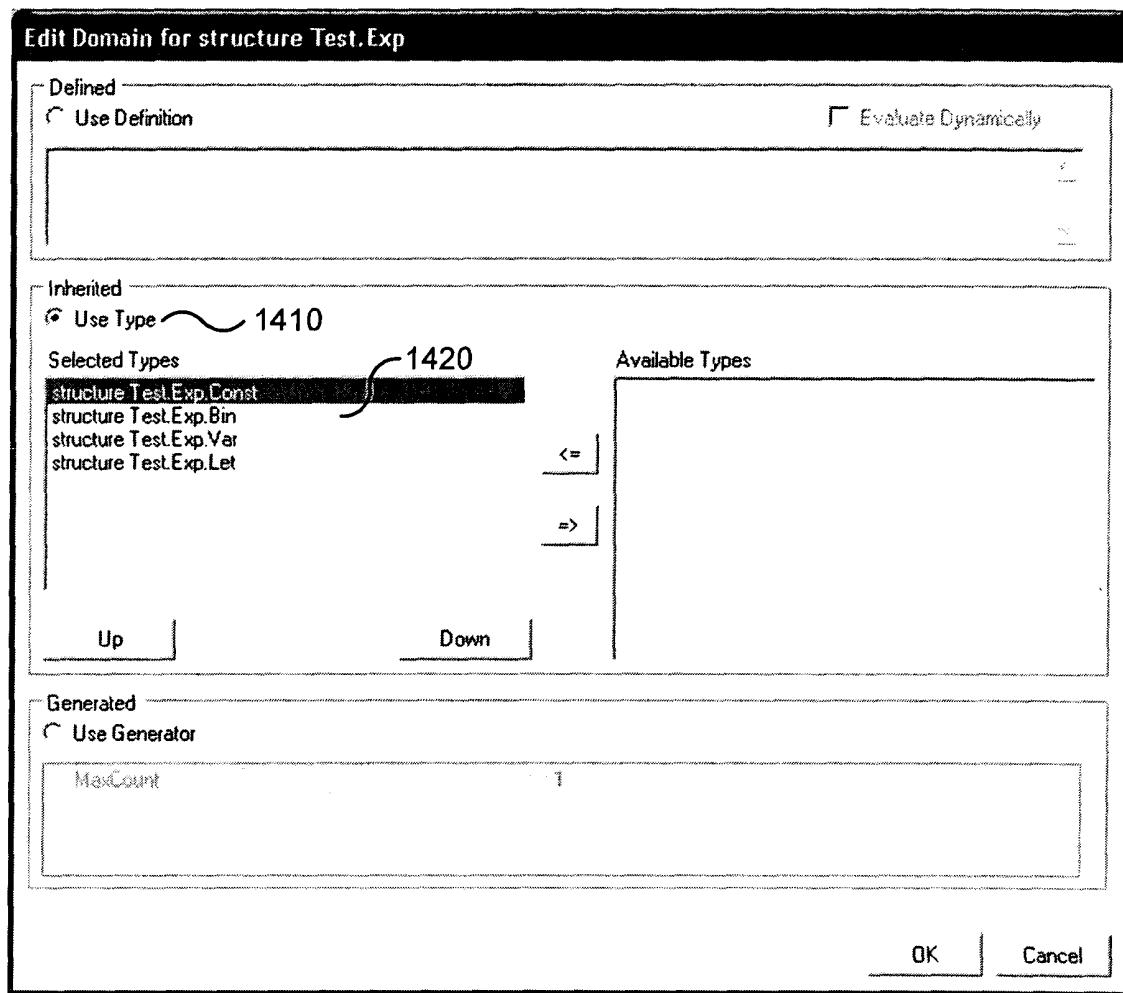


FIG. 15

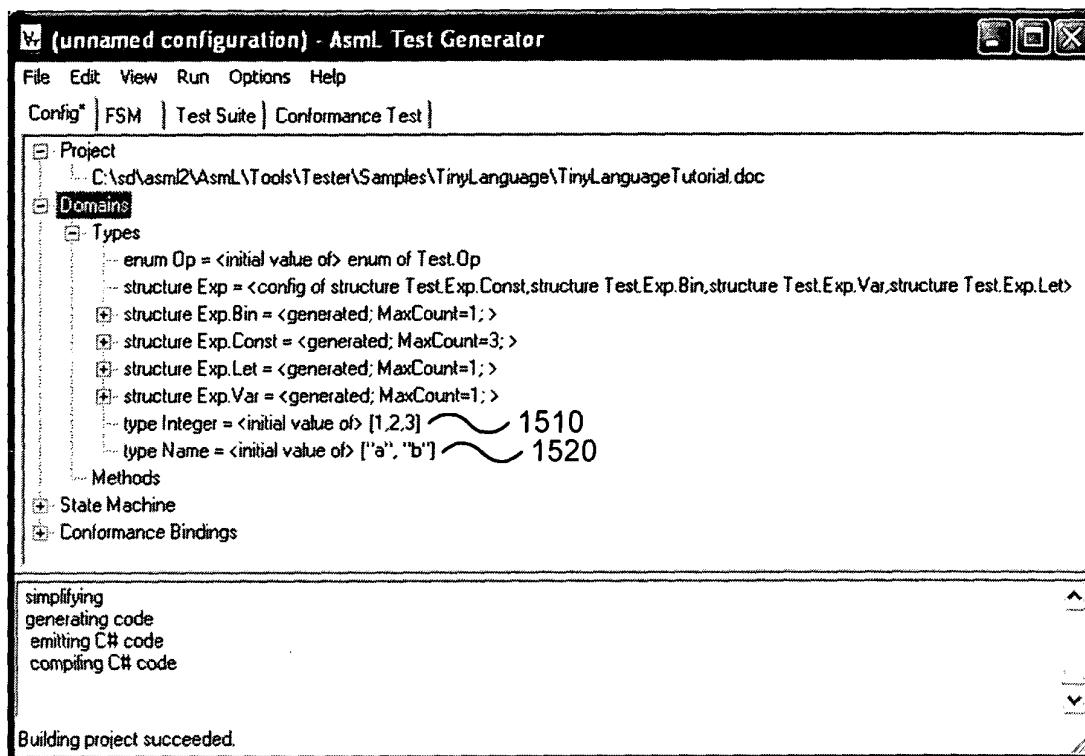


FIG. 16

